What’s an algorithm?

“A procedure for solving a mathematical problem (as of finding the greatest common divisor) in a finite number of steps that frequently involves repetition of an operation.” — Merriam-Webster

“An algorithm is a finite, definite, effective procedure, with some input and some output.” — Donald Knuth
What is this course about?

• In CS230, you learnt how to:
  • Abstract functionality from design
  • Design efficient data structures
  • Design modular applications

• You used to,
  • Code all of that
  • Test it to see if it works
  • Then, the fun starts 😊

What is this course about?

• In CS231, you learn the design and analysis of algorithms to solve problems.

• We’ll always focus on three things:
  • How to understand and define a problem
  • How to implement an algorithm that “correctly” solves that problem
  • How to analyze the performance of that algorithm

• The goal is to define correct and efficient algorithms without having to implement and test them.

Remember: You cannot prove correctness by example!
Why take CS231?

Because ... It’s a major requirement!

Really, why?

- **Understanding and Remembering:**
  - Recognize algorithmic techniques used to solve a problem.
  - Identify the correctness, or lack thereof, of an algorithm.

- **Critical Thinking:**
  - Dissect new problems to identify their input and corresponding output.

- **Practical Thinking:**
  - Determine appropriate algorithmic techniques to solve new problems, by relating new problems to ones in their foundation knowledge.
  - Define correct algorithms to solve new problem, and prove their correctness.
  - Critique existing algorithms.
  - Calculate the asymptotic run time complexity of new algorithms.

- **Projects and Research:**
  - Coordinate tasks and collaborate on writing a final paper.
  - Identify high quality scholarly articles, and their contributions.
  - Summarize existing algorithmic research on a topic of their choice
  - Present summary of research to peers, as part of a team.

- **Interpersonal Relationships:**
  - Collaborate with peers on dissecting new problems.
  - Give feedback to peers on their proofs.
  - Take responsibility for work performed as part of a group.
Textbook


Assignments

- **Schedule**
  - Posted on Thursday / Friday
  - Due the following Thursday (at midnight)

- **Submission**
  - Write your assignments in latex, try overleaf.com
  - You will be provided with a Latex template every assignment
    - Template must be used as is
  - Upload soft copy on your CS231 assignment link

- **Proof modules**
  - In some assignments, you will find a problem marked with [Proof-problem]
  - For these problems, you need to carefully formulate and write your arguments for the correctness of your solutions.
More on proof modules

• Type the proof using latex

• Submit on the same day as the assignment, but in a separate sheet

• This proof will be graded through the weekend, and you’ll get feedback on it by Monday

• Grading:
  • If the proof is correct and complete, you get full points
  • If edits need to be made, you can resubmit the following Friday
  • Note: you can only resubmit once

Exams

• You’ll have two exams during the semester,
  • 1st exam on Oct 21st
  • 2nd exam on Nov 25th

• All exams are in-class and open book

• There will be no final exam.
  • Instead, there will be a final short paper and presentation.
More ...

• All submissions will be done using Gradescope.
  • You’ll get an email this week.

• Assignments must be in PDF format, using Latex.
  • You can use overleaf.com

• Late submissions are not allowed, but you get free passes to use throughout the semester.

• All communication will be done using Piazza.
  • You’ll get an email this week.

CS231 course webpage

• It contains all course info
  • The schedule for the semester
  • Office hours discussion sections and help room
  • The course syllabus

• Make sure you check it often

• Let’s take a quick look
Now, let’s review some data structures

Remember arrays and linked lists? 😊